01 Jan 2006

Intelligent Optimal Control of Excitation and Turbine Systems in Power Networks

Ganesh K. Venayagamoorthy
Missouri University of Science and Technology

Ronald G. Harley

Intelligent Optimal Control of Excitation and Turbine Systems in Power Networks

G. K. Venayagamoorthy, *Senior Member, IEEE*, and R. G. Harley, *Fellow, IEEE*

Abstract--**The increasing complexity of the modern power grid highlights the need for advanced modeling and control techniques for effective control of excitation and turbine systems. The crucial factors affecting the modern power systems today is voltage control and system stabilization during small and large disturbances. Simulation studies and real-time laboratory experimental studies carried out are described and the results show the successful control of the power system excitation and turbine systems with adaptive and optimal neurocontrol approaches. Performances of the neurocontrollers are compared with the conventional PI controllers for damping under different operating conditions for small and large disturbances.**

***Index Terms*— Adaptive Critic Designs, Approximate Dynamic Programming, Excitation Control, Neural Networks, Optimal Control, Reinforcement Learning, Turbine Control.**

I. INTRODUCTION

POWER system control essentially requires a continuous balance between electrical power generation and a varying load demand, while maintaining system frequency, voltage levels and the power grid security. However, generator and grid disturbances can vary between minor and large imbalances in mechanical and electrical generated power, while the characteristics of a power system change significantly between heavy and light loading conditions, with varying numbers of generator units and transmission lines in operation at different times. The result is a highly complex and non-linear dynamic electric power grid with many operational levels made up of a wide range of energy sources with many interaction points. As the demand for electric power grows closer to the available sources, the complex systems that ensure the stability and security of the power grid are pushed closer to their edge. Thus, the need for advanced modeling and control techniques for the effective control of power system elements.

Adaptive critic designs (ACDs) are neural network designs capable of optimization over time, under conditions of noise and uncertainty. This family of ACDs brings new optimization techniques which combine concepts of reinforcement learning and approximate dynamic programming, thus making them powerful tools. The adaptive critic method provides a methodology for designing optimal nonlinear controllers using neural networks for complex systems such as the power system where accurate models are difficult to derive.

This paper describes the work of the authors based on adaptive critics for designing power system stabilization, excitation and turbine neurocontrollers for generators [1]-[3] which overcome the risk of instability [4], the problem of residual error in the system identification [5], input uncertainties [6], and the computational load of online training. The neurocontroller augments/replaces the conventional PI controllers, and is trained in an offline mode prior to commissioning. Two different types of Adaptive Critics are discussed, namely the Heuristic Dynamic Programming (HDP) type and the Dual Heuristic Programming (DHP) type. Results are presented for a single-machine-infinite-bus and a multimachine power system.

II. ADAPTIVE CRITIC DESIGNS

A. Background

The simplest adaptive critic designs learn slowly on large problems but they are successful on many real world difficult small problems. Complex adaptive critics may seem breathtaking, at first, but they are the only design approach that shows potential of replicating critical aspects of human intelligence: ability to cope with a large number of variables in parallel, in real time, in a noisy nonlinear non-stationary environment.

A family of ACDs was proposed by Werbos [7] as a new optimization technique combining concepts of reinforcement learning and approximate dynamic programming. For a given series of control actions that must be taken sequentially, and not knowing the effect of these actions until the end of the sequence, it is impossible to design an optimal controller using the traditional supervised learning neural network. The adaptive critic method determines optimal control laws for a system by successively adapting two ANNs, namely an ***action neural network*** (which dispenses the control signals) and a ***critic neural network*** (which 'learns' the desired performance index for some function associated with the performance index). These two neural networks approximate the Hamilton-Jacobi-Bellman equation associated with optimal control theory. The adaptation process starts with a non-optimal, arbitrarily chosen, control by the action network; the

The support from the National Science Foundation under the grants - CAREER ECS # 0348221 and ECS # 0080764 is gratefully acknowledged by the authors.

G. K. Venayagamoorthy is with the Real-Time Power and Intelligent Systems (RTPIS) Laboratory, Department of Electrical and Computer Engineering, University of Missouri Rolla, MO 65409, USA
(e-mail: gkumar@ieee.org).

R. G. Harley is with School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA (e-mail: ron.harley@ece.gatech.edu).

critic network then guides the action network towards the optimal solution at each successive adaptation. During the adaptations, neither of the networks need any 'information' of an optimal trajectory, only the desired cost needs to be known. Furthermore, this method determines optimal control policy for the entire range of initial conditions and needs no external training, unlike other neurocontrollers.

Dynamic programming prescribes a search which tracks backward from the final step, retaining in memory all suboptimal paths from any given point to the finish, until the starting point is reached. The result of this is that the procedure is too computationally expensive for most real problems. In supervised learning, an ANN training algorithm utilizes a desired output and, having compared it to the actual output, generates an error term to allow the network to learn. The backpropagation algorithm is typically used to obtain the necessary derivatives of the error term with respect to the training parameters and/or the inputs of the network. However, backpropagation can be linked to reinforcement learning via the critic network which has certain desirable attributes.

The technique of using a critic, removes the learning process one step from the control network (traditionally called the "action network" or "actor" in ACD literature), so the desired trajectory is not necessary. The critic network learns to approximate the ***cost-to-go*** or strategic utility function (the function J of Bellman's equation in dynamic programming) and uses the output of the action network as one of its inputs, directly or indirectly.

Different types of critics have been proposed. For example, Watkins [8] developed a system known as Q-learning, explicitly based on dynamic programming. Werbos, on the other hand, developed a family of systems for approximating dynamic programming [7]; his approach subsumes other designs for continuous domains. For example, Q-learning becomes a special case of Action-Dependent Heuristic Dynamic Programming (ADHDP), which is a critic approximating the J function (see section B below), in Werbos' family of adaptive critics. A critic which approximates only the derivatives of the function J with respect to its states, called the Dual Heuristic Programming (DHP), and a critic approximating both J and its derivatives, called the Globalized Dual Heuristic Programming (GDHP), complete this ACD family. These systems do not require exclusively neural network implementations, since any differentiable structure is suitable as a building block. The interrelationships between members of the ACD family have been generalized and explained in detail by Prokhorov [9, 10].

B. Heuristic Dynamic Programming

Fig. 1 shows a model dependent HDP Critic/Action design. The HDP Critic neural network is connected to the Action neural network through a Model neural network of the plant. These three different neural networks used in this study are three-layer feedforward neural networks with a single hidden layer with sigmoidal transfer functions. The input and output layers have linear transfer functions.

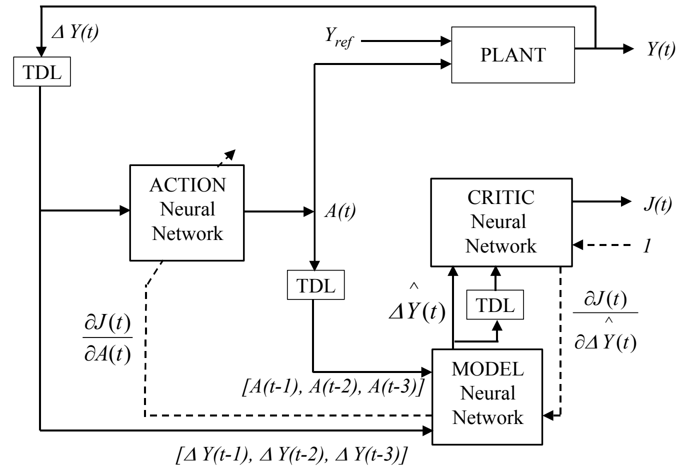

Fig. 1 A model dependent HDP critic/action design.

For model dependent designs it is assumed that there exists a Model neural network which is able to predict the changes in the states/outputs *Y(t+1)*, of the plant at time *t+1*, given at time *t*, the states/outputs, *Y(t)* and the action signals, *A(t)*.

$$\hat{Y}(t+1) = f(Y(t), A(t)) \tag{1}$$

In addition to the signals at time *t*, delayed values of these signals can be used depending on the complexity of the plant dynamics [11]. The inputs to the Model network are time-delayed values (TDL) of both the plant and the Action network outputs. The details on the development of Model networks using supervised learning are explained in [11, 16].

Heuristic Dynamic Programming has a Critic neural network that estimates the function J (cost-to-go) in the Bellman equation of dynamic programming, expressed as follows:

$$J(Y(t)) = \sum_{k=0}^{\infty} \gamma^k U(Y(t+k)) \tag{2}$$

where γ is a discount factor for finite horizon problems ($0 < \gamma < 1$), *U(.)* is the utility function or the local cost and *Y(t)* is an input vector to the Critic. The Critic neural network is trained forward in time (multi-time steps ahead), which is of great importance for real-time operation.

Fig. 2 shows the HDP Critic adaptation/training. The inputs to the Critic are outputs from the Model neural network and its time-delayed values (Fig. 1). Two Critic neural networks are shown in Fig. 2 having the same inputs and outputs but at different time instants. The first Critic neural network has inputs from time steps *t*, *t-1* and *t-2*, and the second Critic neural network has inputs from time steps *t+1*, *t* and *t-1*. Their corresponding outputs are *J(t)* and $\hat{J}(t+1)$ respectively. The second Critic neural network estimates the function $\hat{J}$ (cost-to-go) at time *t+1* by using the Model neural network to get inputs one step ahead. As a result it is possible to know the Critic neural network output $\hat{J}(t+1)$ at time *t*.

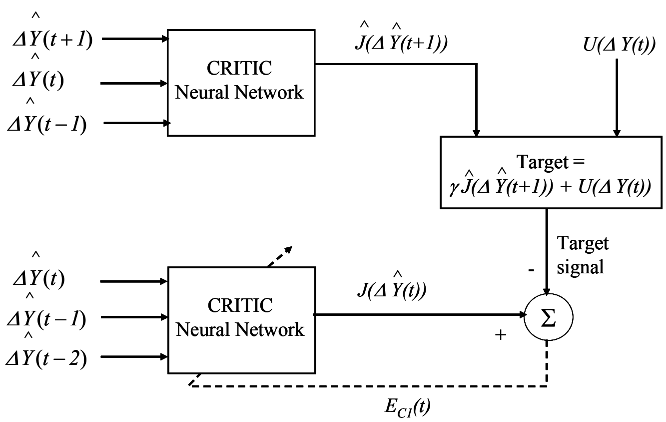

Fig. 2 HDP Critic neural network adaptation/training.

The Critic network tries to minimize the following error measure over time

$$\|E_1\| = \frac{1}{2}\sum_t E_{C1}^2(t) \tag{3}$$

$$E_{C1}(t) = \hat{J}(\Delta\hat{Y}(t)) - \gamma\hat{J}(\Delta\hat{Y}(t+1)) - U(\Delta Y(t)) \tag{4}$$

where $\Delta Y(t)$ is the changes in $Y(t)$, a vector of observables of the plant (or the states, if available). The utility function U is dependent on the system controlled and a typical function is given in [2]. It should be noted that only for the purposes of this study, changes in the state variables are used rather than state variables. The weights' update for the Critic network using the backpropagation algorithm is given as follows:

$$\Delta W_{C1} = -\eta \quad E_{C1}(t)\frac{\partial E_{C1}(t)}{\partial W_{C1}} \tag{5}$$

$$\Delta W_{C1} = -\eta\{J(\Delta\hat{Y}(t)) - \gamma\hat{J}(\Delta\hat{Y}(t+1)) - U(\Delta Y(t))\}\times \frac{\partial\{J(\Delta\hat{Y}(t)) - \gamma\hat{J}(\Delta\hat{Y}(t+1)) - U(\Delta Y(t))\}}{\partial W_{C1}} \tag{6}$$

where η is a positive learning rate and W_{C1} are the weights of the Critic neural network. The same Critic network is shown in two consecutive moments in time in Fig. 2. The Critic network's output $\hat{J}(\Delta\hat{Y}(t+1))$ is necessary in order to provide the training signal $\gamma\hat{J}(\Delta\hat{Y}(t+1)) + U(\Delta Y(t))$, which is the desired/target value for $J(\Delta\hat{Y}(t))$.

The objective of the Action neural network in Fig. 1, is to minimize $J(\Delta Y(t))$ in the immediate future, thereby optimizing the overall cost expressed as a sum of all $U(\Delta Y(t))$ over the horizon of the problem. This is achieved by training the Action neural network with an error signal $\partial J/\partial A$. The gradient of the cost function J, with respect to the outputs A, of the Action neural network, is obtained by backpropagating $\partial J/\partial J$ (i.e. the constant *1*) through the Critic neural network and then through the pretrained Model neural network to the Action neural network. This gives $\partial J/\partial A$ and $\partial J/\partial W_A$ for all the outputs of the Action neural network, and all the Action neural network's weights W_A, respectively. The weights' update in the Action neural network using backpropagation algorithm is given as follows:

$$\|E_2\| = \frac{1}{2}\sum_t E_{A1}^2(t) \tag{7}$$

where
$$E_{A1} = \frac{\partial J(t)}{\partial A(t)} \tag{8}$$

and
$$\frac{\partial J(t)}{\partial A(t)} = \frac{\partial J(t)}{\partial\Delta\hat{Y}(t)}\frac{\partial\Delta\hat{Y}(t)}{\partial A(t)} \tag{9}$$

Weight change in the Action network ΔW_{A1} can be written as:

$$\Delta W_{A1} = -\alpha\frac{\partial J(t)}{\partial A(t)}\frac{\partial}{\partial W_{A1}}\left(\frac{\partial J(t)}{\partial A(t)}\right) \tag{10}$$

where α is a positive learning rate.

With (6) and (10), the training of the Critic and the Action networks can be carried out. The general training procedure for the Critic and the Action networks are described in [1].

C. Dual Heuristic Programming

The Critic neural network in the DHP scheme shown in Fig. 3, estimates the derivatives of J with respect to the vector $\Delta\hat{Y}$ (outputs of the Model neural network) and learns minimization of the following error measure over time:

$$\|E_3\| = \sum E_{C2}^T(t)E_{C2}(t)s \tag{11}$$

where

$$E_{C2}(t) = \frac{\partial J(\Delta\hat{Y}(t))}{\partial\Delta\hat{Y}(t)} - \gamma\frac{\partial\hat{J}(\Delta\hat{Y}(t+1))}{\partial\Delta Y(t)} - \frac{\partial U(\Delta Y(t))}{\partial\Delta Y(t)} \tag{12}$$

and $\partial(.)/\partial\Delta Y(t))$ is a vector containing partial derivatives of the scalar (.) with respect to the components of the vector ΔY. The Critic neural network's training is more complicated than in HDP, since there is a need to take into account all relevant pathways of backpropagation as shown in Fig. 3, where the paths of derivatives and adaptation of the Critic are depicted by dashed lines. In Fig. 3, the dashed lines mean the first backpropagation and the dotted-dashed lines mean the second backpropagation.

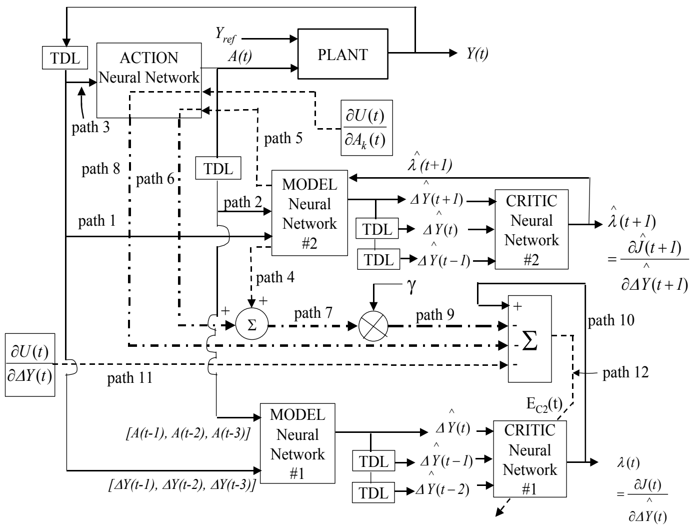

Fig. 3 DHP Critic neural network adaptation.

The Model neural network in the design of DHP Critic and Action neural networks is obtained in a similar manner to that described in [16].

In the DHP scheme, application of the chain rule for derivatives yields:

$$\frac{\partial \hat{J}(\Delta \hat{Y}(t+1))}{\partial \Delta Y_j(t)} = \sum_{i=1}^{n} \hat{\lambda}_i(t+1)\frac{\partial \Delta \hat{Y}_i(t+1)}{\partial \Delta Y_j(t)} + \sum_{k=1}^{m}\sum_{i=1}^{n} \hat{\lambda}_i(t+1)\frac{\partial \Delta \hat{Y}_i(t+1)}{\partial A_k(t)}\frac{\partial A_k(t)}{\partial \Delta Y_j(t)} \tag{13}$$

where $\hat{\lambda}_i(t+1) = \partial \hat{J}(\Delta \hat{Y}(t+1)) / \partial \Delta \hat{Y}_i(t+1)$, and *n, m, j* are the numbers of outputs of the Model, Action and Critic neural networks respectively. By exploiting (13), each of *n* components of the vector $E_{C2}(t)$ from (12) is determined by

$$E_{C2j}(t) = \frac{\partial J(\Delta \hat{Y}(t))}{\partial \Delta \hat{Y}_j(t)} - \gamma \frac{\partial \hat{J}(\Delta \hat{Y}(t+1))}{\partial \Delta Y_j(t)} - \frac{\partial U(\Delta Y(t))}{\partial \Delta Y_j(t)} - \sum_{k=1}^{m}\frac{\partial U(t)}{\partial A_k(t)}\frac{\partial A_k(t)}{\partial \Delta Y_j(t)} \tag{14}$$

The partial derivatives of the utility function *U(t)* with respect to $A_k(t)$, and $\Delta Y(t)$, $\partial U(t)/\partial A_k(t)$ and $\partial U(t)/\partial \Delta Y(t)$ respectively, are obtained by backpropagating the utility function, *U(t)* through the Model network. The adaptation of the action network in Fig. 3, is illustrated in Fig. 4 which propagates $\lambda(t+1)$ back through the model network to the action network. The goal of such adaptation can be expressed as follows [9, 10]:

$$\frac{\partial U(\Delta Y(t))}{\partial A(t)} + \gamma \frac{\partial \hat{J}(\Delta \hat{Y}(t+1))}{\partial A(t)} = 0 \quad \forall \quad t \tag{15}$$

The error signal for the Action network adaptation is therefore given as follows:

$$E_{A2}(t) = \frac{\partial U(\Delta Y(t))}{\partial A(t)} + \gamma \frac{\partial \hat{J}(\Delta \hat{Y}(t+1))}{\partial A(t)} \tag{16}$$

The weights' update expression [10], when applying backpropagation, is as follows:

$$\Delta W_{A2} = -\alpha \left[\frac{\partial U(\Delta Y(t))}{\partial A(t)} + \gamma \frac{\partial \hat{J}(\Delta \hat{Y}(t+1))}{\partial A(t)} \right]^T \frac{\partial A(t)}{\partial W_{A2}} \tag{17}$$

where α is a positive learning rate and W_{A2} are weights of the DHP Action neural network.

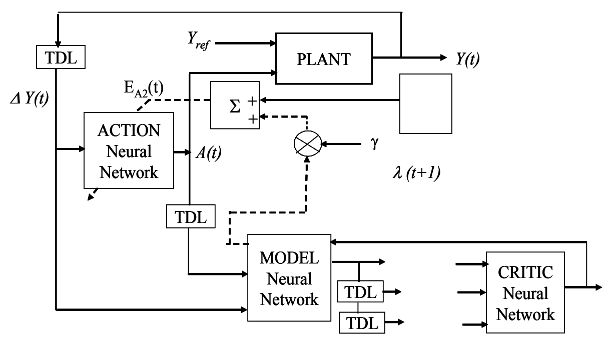

Fig. 4 DHP Action neural network adaptation.

III. ACD Based Control Of Excitation And Turbine Systems Of Generators

The micro-machine laboratory at the University of Kwa-Zulu Natal, Durban, South Africa has two 3 kW, 220 V, three phase micro-alternators, and each one represents both the electrical and mechanical aspects of a typical 1000 MW alternator. The laboratory power system is simulated in the MATLAB/SIMULINK environment and simulations studies with neurocontrollers are carried out prior to hardware implementations. The laboratory single machine infinite bus power system in Fig. 5 consists of a micro-alternator, driven by a dc motor whose torque - speed characteristics are controlled by a power electronic converter to act as a micro-turbine, and a single short transmission line which links the micro-alternator to a voltage source which has a constant voltage and frequency, called an infinite bus. The parameters of the micro-alternators, determined by the IEEE standards are given in [13]. A time constant regulator is used to insert negative resistance in series with the field winding circuit [13], in order to reduce the actual field winding resistance to the correct per-unit value.

A three-machine power system shown in Fig. 6 is set up by using the two micro-alternators and the infinite bus as the third machine.

A. Conventional Excitation and Turbine Control

The practical system uses a conventional AVR and exciter combination of which the transfer function block diagram is shown in Fig. 7, and the time constants and gain are given in [13]. The exciter saturation factor S_e is given by

$$S_e = 0.6093\exp(0.2165V_{fd}) \tag{18}$$

T_{v1}, T_{v2}, T_{v3} and T_{v4} are the time constants of the PID voltage regulator compensator; T_{v5} is the input filter time constant; T_e is the exciter time constant; K_{av} is the AVR gain; V_{fdm} is the exciter ceiling voltage; and, V_{ma} and V_{mi} are the AVR maximum and minimum ceiling voltages.

The block diagram of the power system stabilizer (PSS) used to achieve damping of the system oscillations is shown in Fig. 8 [14]. The considerations and procedures used in the selection of the PSS parameters are similar to that found in [14].

A separately excited 5.6 kW thyristor controlled dc motor is used as a prime mover, called the micro-turbine, to drive the micro-alternator. The torque-speed characteristic of the dc motor is controlled to follow a family of rectangular hyperbola to emulate the different positions of a steam valve, as would occur in a real typical high pressure (HP) cylinder turbine. The three low pressure (LP) cylinders' inertia are represented by appropriately scaled flywheels attached to the micro-turbine shaft. The micro-turbine and governor combination transfer function block diagram is shown in Fig. 9, where, P_{ref} is the turbine input power set point value, P_m is the turbine output power, and $\Delta\omega$ is the speed deviation from the synchronous speed. The turbine and governor time constants and gain are given in [13].

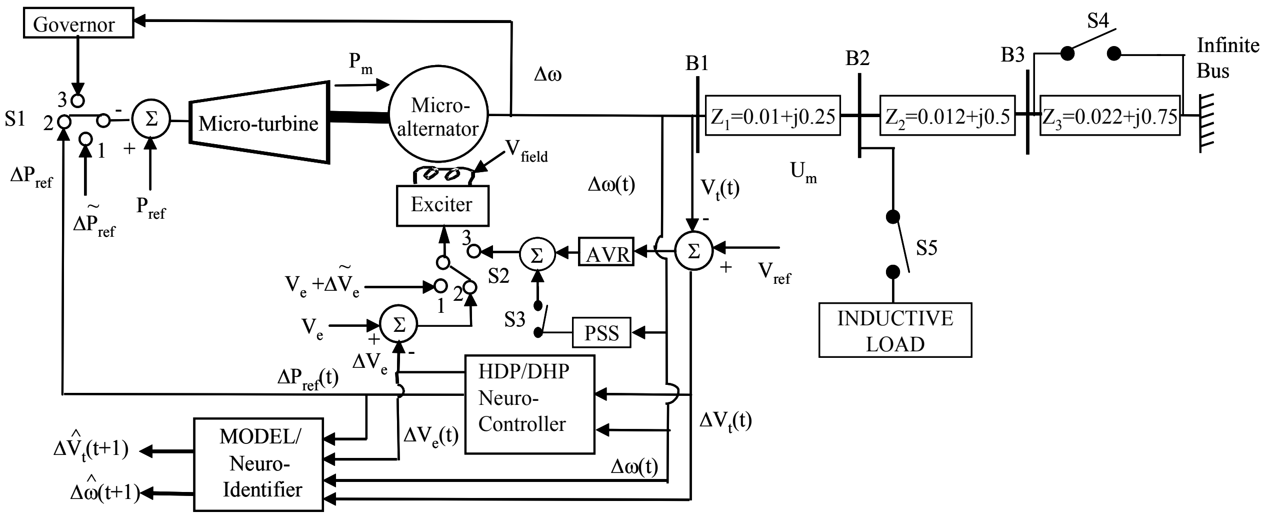

Fig. 5 The single machine infinite bus configuration with the conventional AVR and governor controllers, and neurocontroller.

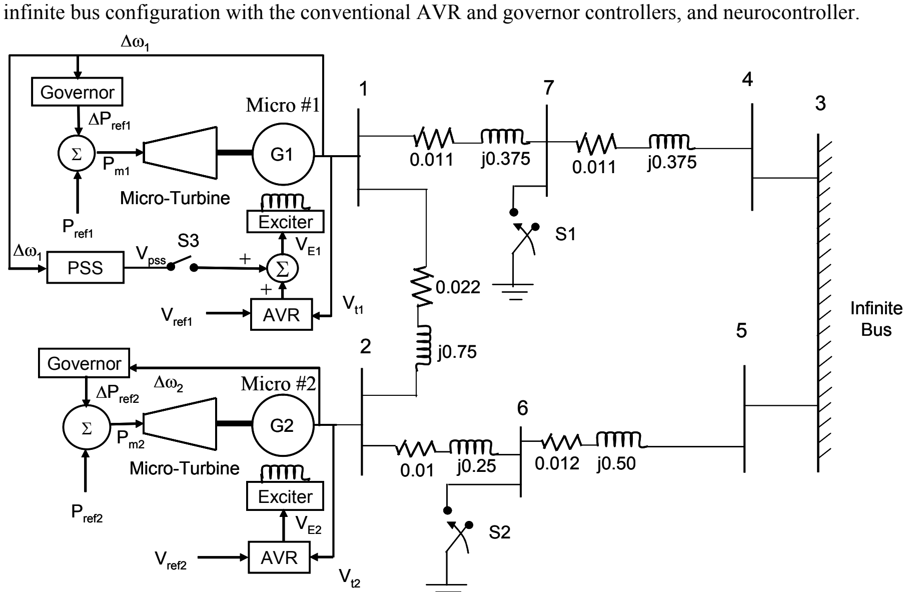

Fig. 6 Multimachine power system consisting of two micro-alternators G1 and G2 which are conventionally controlled by the AVRs, governors and a PSS.

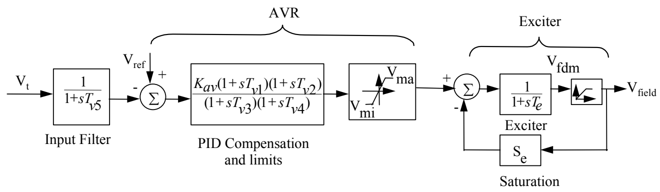

Fig. 7 Block diagram of the AVR and exciter combination.

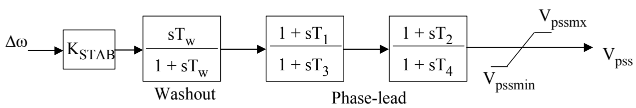

Fig. 8 Block diagram of the power system stabilizer.

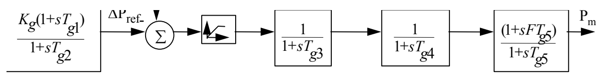

Fig. 9 Block diagram of the micro-turbine and governor combination.

The gains K_{av} (0.003) of the AVR and K_g (0.05) of the governor are obtained by suitable choices of the gain and phase margins in each case, as described in [15]. Transmission lines are represented by using banks of lumped inductors and capacitors.

B. Simulation and Experimental Studies with Different Control Schemes for Excitation and Turbine Systems

The dynamic and transient operation of the HDP and DHP neurocontrollers is compared with the operation of the conventional (CONV) controller (AVR and turbine governor, excluding the PSS) for single machine infinite bus power system in Fig. 5. In addition, the performance of a continually online trained neurocontroller (COT) is also shown. The COT neurocontroller is developed based on the indirect adaptive neurocontrol scheme [16]. In power systems faults such as three phase short circuits occur from time to time, and because they prevent energy from the generator reaching the infinite bus, it means that most of the turbine shaft power goes into accelerating the generator during the fault. This represents a severe transient test for the controller performance. Figs. 10 and 11 show the response of all four controllers for the three phase temporary short circuit for 50 ms with the new transmission line impedance Z_2. Here, it is obvious that the DHP controller clearly beats the other three controllers in terms of offering the greatest oscillation damping especially in the rotor angle. The DHP controller proves its robustness to changes in the system configurations.

Based on the results for the single machine power system above, the DHP controller has the best performance; hence, the DHP neurocontroller is the only one that is now implemented on the multimachine power system. The performance of the DHP neurocontroller is now compared with that of the conventional controllers, one of which is equipped with a power system stabilizer. Fig. 12 shows the multimachine power system of Fig. 6 now equipped with two DHP neurocontrollers.

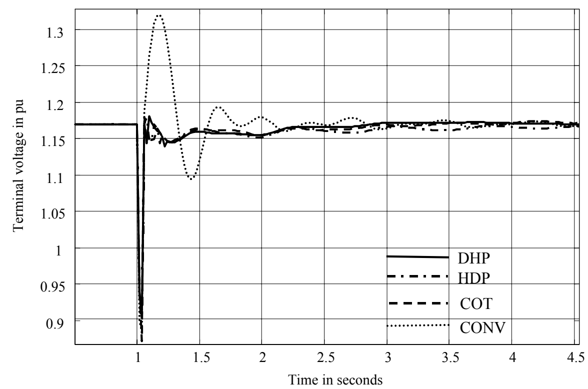

Fig. 10 Terminal voltage of the micro-alternator for a temporary 50 ms three phase short circuit (transmission line impedance Z_2).

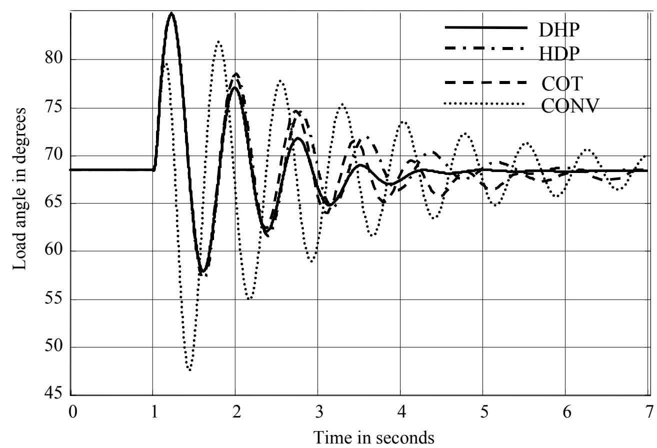

Fig. 11 Rotor angle of the micro-alternator for a temporary 50 ms three phase short circuit (transmission line impedance Z_2).

The DHP neurocontrollers were implemented on DSPs and allowed to control the laboratory multimachine power system [3]. The purpose of these tests is to confirm via practical measurements the potential of adaptive critic based neurocontrollers which have been demonstrated during the simulation studies for a single machine and a multimachine power system. However, the laboratory implementation on micro-machines is also intended to form a basis for possible future investigations into use of such neurocontrollers on large multi-megawatt sized power plants in a real-world power station.

At the operating condition *(P = 0.2 pu, Q = 0 pu* on both generators*)*, the series transmission line impedance is increased at time t = 10 s from Z = 0.022 + j0.75 pu to Z = 0.044 + j1.50 pu by opening switch S2. Fig. 13 shows the load angle response of generator G2. The load angle response of generator G1 for the same disturbance is shown in Fig.14.

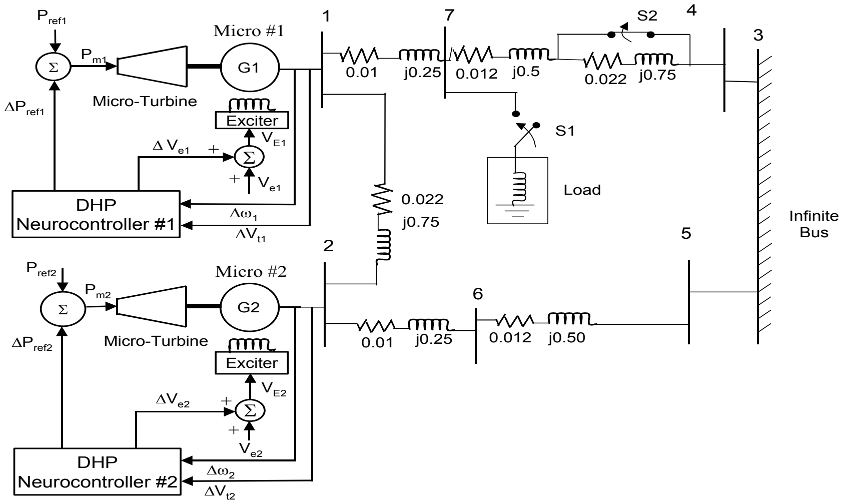

Fig. 12 Multimachine power system with two DHP neurocontrollers.

Four different controller combination studies are carried out for the above disturbance.

- Case a - conventional controller on both G1 and G2
- Case b - conventional controller with a PSS on G1 and conventional controller on G2
- Case c - DHP neurocontroller on G1 and conventional controller on G2
- Case d - DHP neurocontrollers on both G1 and G2.

It is clear the DHP neurocontrollers exhibit the best damping of the controllers.

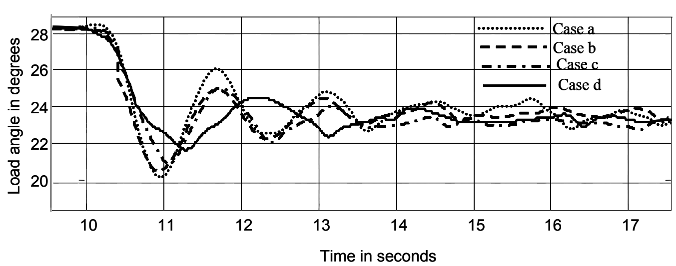

Fig. 13 Load angle response of generator G2 for series transmission line impedance increase by opening switch S2 for P = 0.2 pu and Q = 0 pu.

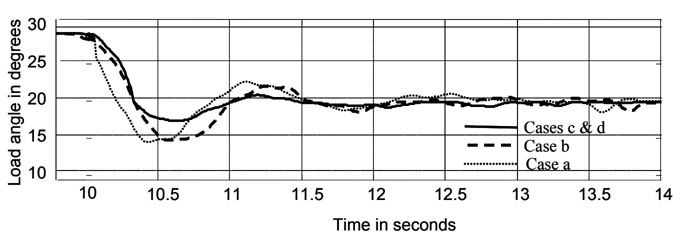

Fig. 14 Load angle response of generator G1 for series transmission line impedance increase by opening switch S2 for P = 0.2 pu and Q = 0 pu.

IV. Conclusion

This paper has presented the investigations on the design and implementation of Adaptive Critic based neurocontrollers to replace/augment the conventional PI controllers on generators in both single-machine-infinite-bus and multimachine power system. These neurocontrollers exhibit better damping than the conventional controllers. The *Adaptive Critic Design* based neurocontrollers have the great advantage that once trained, their weights/parameters remain fixed and therefore avoid the risk of instability associated with continual online training. The convergence guarantee of the Critic and Action neural networks during offline training was shown in [4, 18]. In addition, the heavy computational load of online training only arises during the offline training phase and therefore makes the online real time implementation cost of the neurocontrollers cheaper. The processing hardware cost is a small fraction of the cost of turbogenerators and therefore this is not a big issue.

The *Adaptive Critic Design* based nonlinear optimal controllers designed presented are all based on approximate models obtained by neuroidentifiers, but nevertheless exhibit superior performance in comparison to the conventional linear controllers which use more extensive linearized models. This benefit of a neuroidentifier agrees with the conclusions on the comparison of using approximate and exact models in adaptive critic designs which was explicitly shown in [5]. All these features are desirable and important for industrial applications which require a neurocontroller technology that is nonlinear, robust and stable.

References

[1] G. K. Venayagamoorthy, R. G. Harley, D. C. Wunsch, "Comparison of Heuristic Dynamic Programming and Dual Heuristic Programming Adaptive Critics for Neurocontrol of a Turbogenerator", *IEEE Transactions on Neural Networks*, Volume: 13, Issue: 3, May 2002, Page(s): 764 -773.

[2] G. K. Venayagamoorthy, R. G. Harley, D. C. Wunsch, "Dual Heuristic Programming Excitation Neurocontrol for Generators in a Multimachine Power System", *IEEE Transactions on Industry Applications*, Volume: 39, Issue: 2, March/April 2003.

[3] G. K. Venayagamoorthy, R. G. Harley, D. C. Wunsch, "Implementation of Adaptive Critic Based Neurocontrollers for Turbogenerators in a Multimachine Power System", *IEEE Transactions on Neural Networks –Special Issue on Hardware Implementations*, September 2003.

[4] D. Prokhorov, L. A. Feldkamp, "Analyzing for Lyapunov Stability with Adaptive Critics", *Proceedings of the International Conference on Systems, Man and Cybernetics*, 1998, Vol. 2, pp. 1658-1161.

[5] T. T. Shannon, G. G. Lendaris, "Qualitative Models for Adaptive Critic Neurocontrol", *Proceedings of the International Joint Conference on Neural Networks*, IJCNN 1999, Washington DC, USA, Vol. 1, pp. 455-460.

[6] Z. Huang, S. N. Balakrishnan, "Robust Adaptive Critic Based Neurocontrollers for Systems with Input Uncertainties", *Proceedings of the International Joint Conference on Neural Networks*, *(IJCNN 2000),* 24 – 27 July, 2000, Como, Italy, Vol. 3, pp. 67-72.

[7] P. J. Werbos, "Approximate Dynamic Programming for Real Time Control and Neural Modelling", in White DA and Sofge DA (Eds.), *Handbook of Intelligent Control,* Van Nostrand Reinhold, New York, 1992, ISBN 0-442-30857-4, pp. 493 - 525.

[8] C. J. Watkins, P. Dayan, "Q-Learning", *Machine Learning*, vol. 8, 1992, pp. 279 – 292.

[9] D. V. Prokhorov, "Adaptive Critic Designs and their Applications", *Ph.D. Thesis*, Texas Tech University, USA, October 1997.

[10] D. V. Prokhorov, D. C. Wunsch, "Adaptive Critic Designs" *IEEE Trans. on Neural Networks*, Vol. 8, No. 5, September 1997, pp. 997 – 1007.

[11] G. K. Venayagamoorthy, R. G. Harley, " A Continually Online Trained Artificial Neural Network Identifier for a Turbogenerator", *Proceedings of IEEE International Electrical Machine and Drives Conference (IEMDC)*, Seattle, USA, May 1999, pp. 404 – 406.

[12] K. S. Narendra, K. Parthasarathy, "Identification and Control of Dynamical Systems using Neural Networks", *IEEE Trans. on Neural Networks*, Vol.1, No. 1, March 1990, pp. 4-27.

[13] D. J. Limebeer, R. G. Harley, S. M. Schuck, "Subsychronous Resonance of the Koeberg Turbogenerators and of a Laboratory Micro-Alternator System", *Trans. of the SA Institute of Electrical Engineers*, November 1979, pp. 278-297.

[14] P. Kundur, M. Klein, G. J. Rogers, M. S. Zywno, "Application of Power System Stabilizers for Enhancement of Overall System Stability", *IEEE Trans. on Power Systems*, Vol. 4, No. 2, May 1989, pp. 614 – 626.

[15] W. K. Ho, C. C. Hang, L. S. Cao, "Tuning of PID Controllers based on Gain and Phase Margin Specifications", *Proceedings of the 12th Triennial World Congress on Automatic Control*, 1993, pp. 199 –202.

[16] G. K. Venayagamoorthy, R. G. Harley, "A Continually Online Trained Neurocontroller for Excitation and Turbine Control of a Turbogenerator", *IEEE Trans. on Energy Conversion*, Vol. 16, No.3, September 2001, pp. 261-269.

[17] P. M. Anderson, A. A. Fouad, *Power system control and stability.* New York: IEEE Press, 1994.

[18] D. V. Prokhorov, D. C. Wunsch, "Convergence of Critic-Based Training", *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, Vol. 4, 1997, pp. 3057 – 3060.

Ganesh Kumar Venayagamoorthy (S'91, M'97, SM'02) received his PhD degree in Electrical Engineering from the University of Natal, Durban, South Africa, in February 2002. He is currently an Assistant Professor of Electrical and Computer and the Director of the Real-Time Power and Intelligent Systems Laboratory at University of Missouri, Rolla. His research interests are in computational intelligence, power systems control and stability, evolvable hardware and signal processing. He has published over 150 papers in refereed journals and international conferences. Dr. Venayagamoorthy is the recipient, of the following awards - 2005 IEEE Industry Application Society (IAS) Outstanding Young Member award, 2005 South African Institute of Electrical Engineers Young Achiever's award, 2004 NSF CAREER award, the 2004 IEEE St. Louis Section Outstanding Young Engineer award, the 2003 International Neural Network Society (INNS) Young Investigator award, 2001 IEEE Computational Intelligence Society (CIS) W. J. Karplus summer research grant and five prize papers with the IEEE IAS and IEEE CIS. He is a Senior Member of the IEEE and the South African Institute of Electrical Engineers, a Member of INNS and the American Society for Engineering Education. He is an Associate Editor of the IEEE Transactions on Neural Networks and was a Guest Editor for the Neural Networks journal. He is currently the IEEE St. Louis IAS Chapter Chair, the Chair and the founder of IEEE St. Louis CIS Chapter, the Chair of the Task Force on Intelligent Control Systems and the Secretary of the Intelligent Systems subcommittee of IEEE Power Engineering Society. Dr. Venayagamoorthy was the Technical Program Co-Chairs of the 2003 International Joint Conference on Neural Networks, Portland, OR, USA and the 2004 International Conference on Intelligent Sensing and Information Processing, Chennai, India. He has served as member of the program committee, organized and chaired panel/special sessions, and presented tutorials at several international conferences and workshops.

Ronald G. Harley (M'77-SM'86-F'92) was born in South Africa. He received the BSc.Eng. (cum laude) and MScEng (cum laude) degrees from the University of Pretoria, Pretoria, South Africa, and the Ph.D. degree from London University, London, U.K., in 1960, 1965, and 1969, respectively. In 1970 he was appointed to the Chair of Electrical Machines and Power Systems at the University of Natal in Durban, South Africa. He is currently at the Georgia Institute of Technology, Atlanta, USA. He has co-authored some 350 papers in refereed journals and international conferences. Altogether 9 papers attracted prizes from journals and conferences. Ron is a Fellow of the SAIEE, a Fellow of the IEE, and a Fellow of the IEEE. He is also a Fellow of the Royal Society in South Africa, a Fellow of the University of Natal, and a Founder Member of the Academy of Science in South Africa formed in 1994. He has been elected as a Distinguished Lecturer by the IEEE Industry Applications Society for the years 2000 and 2001. His research interests are in the dynamic and transient behavior of electric machines and power systems, and controlling them by the use of power electronics and modern control algorithms.